REMARKS

Reconsideration of the above-identified patent application in view of the amendments above and the remarks following is respectfully requested.

Claims 1, 3-9, 11, 12, 14, 16-31, 33-39, 41, 42, 44, 46-59 and 64-66 are in this case. Claims 20-30 and 50-59 have been withdrawn by the Examiner from consideration as drawn to a non-elected invention. Claims 1, 3-9, 11, 12, 14, 16-19, 31, 33-39, 41, 42, 44, 46-49 and 64-66 have been rejected under § 103(a). Dependent claims 3 and 33 have been canceled. Independent claims 1 and 31 have been amended.

Specifically:

In claim 1, the recitation of the outgoing packet generator comprising a gather engine has been moved to the end of the claim, for clarity, without the limitation that the write data and the read data are gathered via a commonly shared data flow path.

The limitations of claim 3 have been appended to the end of claim 1, and have been amended to make explicit that the gather engine is adapted to read information from both the response descriptor and the request descriptor.

Claim 31 has been amended similarly, including appending to claim 31 the limitations of claim 33. Correspondingly, claims 3 and 33 have been canceled.

An example of the support in the specification for the amendments that make it explicit that the gather engine is adapted to read information from both the response descriptor and the request descriptor starts on page 5 lines 8-26 of the specification:

> By the same token, in generating RDMA write and send requests to a remote responder, as in preparing RDMA read responses to send to a remote requester, the HCA "gathers" data from the local memory and sends it in packets to a remote destination. Client processes on the local host generate write and send requests by submitting WRs to the HCA, so that WQEs are placed in the appropriate HCA queues. A gather engine services the WQEs by

> reading the specified data from the local memory and inserting the data in request packets for transmission. To conform to this model, when the HCA receives RDMA read requests from a remote requester, it similarly generates a list of quasi-WQEs in local memory, which identify the data to be sent to the requester. These quasi-WQEs differ semantically from the WQEs generated by the local host, but they are handled by the HCA in the same way. The quasi-WQEs are serviced by the same gather engine that is responsible for servicing the write and send requests. (emphasis added)

The response descriptors of the preferred embodiments are defined on page 25 lines 6-11 as "quasi-WQEs". Similarly, the WQEs defined on page 19 lines 12-14 are request descriptors. The components of the preferred embodiments that function as a "gather engine" include at least execution engines 92 and SDE 66. How the gather engine services the WQEs and the quasi-WQEs is described on page 22 line30 through page 23 line 3:

> The execution engine parses each WQE and prepares instructions to SDE 66 regarding a request packet or packets to be sent out. (Similarly, for each quasi-WQE, the execution engine prepares instructions to the SDE regarding the required response packet.) (emphasis added)

and on page 26 lines 7-11:

> The execution engine looks up and parses the next quasi-WQE 110 to be executed for the QP in RDB 40, and it instructs SDE 66 to retrieve the data indicated by the quasi-WQE for inclusion in the packets. (emphasis added)

"Looking up and parsing" a WQE or a quasi-WQE is an example of reading information from the WQE or quasi-WQE.

§ 103(a) Rejections – Pettey et al. '712 in view of Gasbarro et al. '004

The Examiner has rejected claims 1, 3-9, 11, 12, 14, 16-19, 31, 33-39, 41, 42, 44, 46-49 and 64-66 under § 103(a) as being obvious over Pettey et al., US Patent No. 6,594,712 (henceforth, "Pettey et al. '712") in view of Gasbarro et al., US Patent No.

6,948,004 (henceforth, "Gasbarro et al. '004"). The Examiner's rejection is respectfully traversed.

Pettey et al. '712 teach an InfiniBand target channel adapter (TCA) **202** of an InfiniBand I/O unit **108** that communicates with a host **102** via an InfiniBand fabric **114**.

Gasbarro et al. '004 teaches a host fabric adapter **120** that a host system **130** uses for communicating with other systems via an InfiniBand switched fabric network **100'**.

Both host system **102** of Pettey et al. '712 and host system **130** of Gasbarro et al. '004 instruct their respective fabric adapters (TCA **202** or host fabric adapter **120**) to send packets to their respective InfiniBand networks (InfiniBand fabric **114** or switched fabric network **100'**) as described in the Background section of the above-identified patent application on page 2 lines 8-15:

> To send and receive communications over the network, the client initiates work requests (WRs), which causes work items, called work queue elements (WQEs), to be placed onto the appropriate queues. The channel adapter then executes the work items, so as to communicate with the corresponding QP of the channel adapter at the other end of the link.

In the case of Pettey et al. '712 (column 11 lines 21-24),

> When the CPU **208** of FIG. **2** desires to send the host **102** a message, it submits a work request **722** to the TCA **202** Send Queue **714**. The TCA **202** creates a Work Queue Entry (WQE) and places the WQE on the Send Queue **714**.

In the case of Gasbarro et al. '004 (column 7 lines 33-38),

> Work requests submitted by a consumer in a form Work Queue Elements "WQEs" are posted onto appropriate work queues (WQs) from the host system **130** to describe data movement operations and location of data to be moved for processing and/or transportation, via the switched fabric **100'**.

See also Gasbarro et al. '004 column 17 lines 49-52:

> "WQEs" are posted onto appropriate work queues (WQs) by the host software of the host system **130** to describe data transfer operations, via the switched fabric **100'**.

These WQEs correspond to the "request descriptors" recited in claims 1 and 33 as now amended. Claims 1 and 33 as now amended distinguish clearly between "request descriptors" and "response descriptors". In claim 1, a request descriptor indicative of the data for the outgoing write request packet is written by the host processor, and a response descriptor indicating data to be read in response to the incoming read request packet is written by the incoming packet processor. In claim 31, a request descriptor indicative of the data for the outgoing write request packet is generated, and a response descriptor indicating data to be read for the outgoing response packet is written in response to the incoming read request packet. In both claim 1 and claim 31, the gather engine gathers both kinds of data in response to both kinds of descriptors.

Neither Pettey et al. '712 nor Gasbarro et al. '004 teach, hint or suggest anything resembling the "response descriptors" recited in claims 1 and 31 as now amended. For example, the way Pettey et al. '712 handle an incoming RDMA Read Request packet, which is an example of an "incoming read request packet", is described in column 14 lines 61-65:

> If the received packet is an RDMA Read Request packet **1200**, then no data is transferred by the RxPP logic **1416**. Instead, the RxPP logic **1416** forwards the received packet to the TxPP logic **1414** for creation of an outgoing RDMA Read Response packet **1300**.

What TxPP logic **1414** does next is described in column 14 lines 25-34:

> The TxPP logic **1414** utilizes SGLs **900** of FIG. **9** to generate the transmit packets from data at local addresses on the PCI buses **212** and **216** of FIG. **2**...The TxPP logic **1414** utilizes a TxPP Scratchpad memory **1404** inside the Bus Router **306** to locally process the SGLs **900** more efficiently.

In other words, Pettey et al. '712 handle incoming read request packets *without creating and responding to a WQE or anything resembling a WQE.* This is as

opposed to the component of TCA **202** that *does* respond to WQEs: Work Queue Management logic **1412**, as described in column 14 lines 12-20. To interpret Pettey et al. '712 as handling an incoming read request packet by writing a response descriptor to a memory outside TCA **202** and then gathering the requested data from local memory **218** in response to the response descriptor, as recited in claims 1 and 31 as now amended, would constitute impermissible hindsight on the part of the Examiner.

Similarly, in their discussion of RDMA operations, Gasbarro et al. '004 note that only the requestor of a RDMA operation, and not the responder of a RDMA operation, posts a WQE. See column 7 lines 47-50:

> For an RDMA operation, the WQE also specifies the address in the remote consumer's memory. Thus an RDMA operation does not need to involve the receive work queue of the destination. (emphasis added)

In his "Response to Arguments" in the Office Action mailed January 17, 2008, in the third paragraph on page 8, the Examiner interpreted Gasbarro et al. '004 column 7 lines 32-67 as teaching

> ...that the Infiniband standard in Gasbarro utilizes the WQE which also specifies the address in the remote consumer's memory for an RDMA operation such as RDMA Write, RDMA Read and Atomic.

Gasbarro et al. '004 column 7 lines 32-67 describe RDMA operations from the point of view of the *requestor*, not from the point of view of the *responder*. Gasbarro et al. '004 distinguish clearly between a "consumer" that requests a RDMA operation and a "remote consumer" that responds to a RDMA operation. Only the consumer posts WQEs, as stated in the above citation from Gasbarro et al. '004 column 7 lines 33-38. The remote RDMA consumer does not post WQEs.

With independent claims 1 and 31 allowable in their present form it follows that claims 3-9, 11, 12, 14, 16-19, 33-39, 41, 42, 44, 46-49 and 64-66 that depend therefrom also are allowable.

Although claims 65 and 66 are allowable merely by virtue of depending from claims 1 and 31, Applicant respectfully presents additional reasons why claims 65 and 66 are allowable. Claims 65 and 66 limit the incoming read request packets of claims 1 and 31 explicitly to RDMA read request packets and the response descriptors of claims 1 and 31 explicitly to quasi-WQEs. In rejecting claims 65 and 66, the Examiner has cited Pettey et al. '712 column 11 lines 1-53 as teaching the recited limitations. Pettey et al. '712 column 11 lines 1-53 teach no such thing. Pettey et al. '712 column 11 lines 1-53 describe TCA **202** acting as a requestor, not as a responder. For example, column 11 lines 21-24, previously cited, state:

> When the CPU **208** of FIG. **2** desires to send the host **102** a message, it submits a work request **722** to the TCA **202** Send Queue **714**. The TCA **202** creates a Work Queue Entry (WQE) and places the WQE on the Send Queue **714**. (emphasis added)

In particular (column 11 lines 31),

> ...RDMA Read WQE **763**...specify, among other things, a virtual address in host **102** memory **124** for data transfers with the I/O unit **108**.

I/O unit **108** is the requestor. Host **102** is the responder. RDMA Read WQE **763** is posted in Send Queue **714** of TCA **202** acting as a requestor, not as a responder. Even Receive WQEs 782 are requestor WQEs, not responder WQEs. As stated in column 11 lines 39-41,

> Receive WQEs **782** are placed on the Receive Queue **716** when the CPU **208** submits a work request **724** to the TCA **202**. (emphasis added)

The Examiner also has cited Gasbarro et al. '004 column 7 lines 33-67 as teaching the limitations recited in claims 65 and 66. Gasbarro et al. '004 column 7 lines 33-67 teach no such thing. Gasbarro et al. '004 column 7 lines 33-67 describe host system 130 acting as a requestor, not as a responder. For example, column 7 lines 33-38, previously cited, state:

> Work requests submitted by a consumer in a form Work Queue Elements "WQEs" are posted onto appropriate work queues (WQs) from the host system **130** to describe data movement operations and location of data to be moved for processing and/or transportation, via the switched fabric **100'**. (emphasis added)

In particular, with regard to RDMA operations, column 7 lines 47-50, also cited previously, state explicitly that only the requestor of an RDMA operation, and not the responder of an RDMA operation, posts a WQE.

The Examiner also has cited Gasbarro et al. '004 column 13 line 15 to column 14 line 28 as teaching the limitations recited in claims 65 and 66. Gasbarro et al. '004 column 13 line 15 to column 14 line 28 teach no such thing. The Examiner should have begun his citation at column 13 line 6, not column 13 line 15. Column 13 lines 6-14 describe the Send Queue of a Work Queue pair in general terms as an "initiator" (column 13 line 7). Then Column 13 lines 15-19 describe the Receive Queue of a Work Queue pair in general terms as a "responder" (column 13 line 16). Only in column 13 lines 30-35 do Gasbarro et al. '004 say anything about posting WQEs to work queue pairs. As noted above, Gasbarro et al. '004 teach, in column 7 lines 33-38 and in column 17 lines 49-52, that WQEs are posted by host system **130**, and not by an incoming packet processor as recited in claims 1 and 31 from which claims 65 and 66 depend. In particular, as noted above, Gasbarro et al. '004 teach in column 7 lines 47-50 that only the requestor of a RDMA operation such as the RDMA read operation recited in claims 65 and 66, and not the responder of a RDMA operation, posts a WQE.

In view of the above amendments and remarks it is respectfully submitted that independent claims 1 and 31, and hence dependent claims 3-9, 11, 12, 14, 16-19, 33-39, 41, 42, 44, 46-49 and 64-66 are in condition for allowance. Prompt notice of allowance is respectfully and earnestly solicited.

Respectfully submitted,

Mark M. Friedman
Attorney for Applicant
Registration No. 33,883

Date: July 15, 2008